**Rectangle**Problem Code: **RECTANGL**

******

[Submit](https://www.codechef.com/JAN18/submit/RECTANGL)

**Read problems statements in**[**Mandarin chinese**](http://www.codechef.com/download/translated/JAN18/mandarin/RECTANGL.pdf)**,**[**Russian**](http://www.codechef.com/download/translated/JAN18/russian/RECTANGL.pdf)**and**[**Vietnamese**](http://www.codechef.com/download/translated/JAN18/vietnamese/RECTANGL.pdf)**as well.**

You are given four integers **a**, **b**, **c** and **d**. Determine if there's a rectangle such that the lengths of its sides are **a**, **b**, **c** and **d** (in any order).

**Input**

* The first line of the input contains a single integer **T** denoting the number of test cases. The description of **T** test cases follows.
* The first and only line of each test case contains four space-separated integers **a**, **b**, **c** and **d**.

**Output**

For each test case, print a single line containing one string "YES" or "NO".

**Constraints**

* 1 ≤ **T** ≤ 1,000
* 1 ≤ **a**, **b**, **c**, **d** ≤ 10,000

**Subtasks**

**Subtask #1 (100 points):** original constraints

**Example**

**Input:**

3

1 1 2 2

3 2 2 3

1 2 2 2

**Output:**

YES

YES

NO

Author:[6★kingofnumbers](https://www.codechef.com/users/kingofnumbers)

Date Added:2-01-2018

Time Limit:1 secs

Source Limit:50000 Bytes

Languages:ADA, ASM, BASH, BF, C, CAML, CLOJ, CLPS, CPP 4.3.2, CPP 6.3, CPP14, CS2, D, ERL, FORT, FS, GO, HASK, ICK, ICON, JAVA, JS, kotlin, LISP clisp, LISP sbcl, LUA, NEM, NICE, NODEJS, PAS fpc, PAS gpc, PERL, PERL6, PHP, PIKE, PRLG, PYPY, PYTH, PYTH 3.5, RUBY, rust, SCALA, SCM chicken, SCM guile, SCM qobi, ST, swift, TCL, TEXT, WSPC

<https://www.codechef.com/JAN18/problems/RECTANGL>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp19

{

class Program

{

static void Main(string[] args)

{

int t = int.Parse(Console.ReadLine().Trim());

while(t-- > 0)

{

int[] arr = Array.ConvertAll(Console.ReadLine().Trim().Split(), e => int.Parse(e));

Dictionary<int, int> diccio = new Dictionary<int, int>();

for (int i = 0; i < arr.Length; i++)

{

if (diccio.ContainsKey(arr[i]))

{

diccio[arr[i]]++;

}

else

{

diccio[arr[i]] = 1;

}

}

string ans = "YES";

foreach (KeyValuePair<int,int> kvp in diccio)

{

if(kvp.Value != 2 && kvp.Value != 4)

{

ans = "NO";

break;

}

}

Console.WriteLine(ans);

}

Console.ReadLine();

}

}

}